

Structure and Kinematics Triangulation with a Rolling Shutter Stereo Rig

Omar Ait-Aider
LASMEA-CNRS

24 Avenue des Landais 63177 FRANCE

Omar.Ait-Aider@lasmea.univ-bpclermont.fr

François Berry
LASMEA-CNRS

24 Avenue des Landais 63177 FRANCE

Francois.Berry@lasmea.univ-bpclermont.fr

Abstract

We describe a spatio-temporal triangulation method to be used with rolling shutter cameras. We show how a single pair of rolling shutter images enables the computation of both structure and motion of rigid moving objects. Starting from a set of point correspondences in the left and right images, we introduce the velocity and shutter characteristics in the triangulation equations. This results in a non-linear error criterion whose minimization in the least square sense provides the shape and velocity parameters. Unlike previous work on rolling shutter cameras, the constraining assumption of a-priori knowledge about the object geometry is removed and a full 3D motion model is considered. The aim of this work is thus to make the use of rolling shutter cameras of a broader interest. Experimental evaluation results confirm the feasibility of the approach.

1. Introduction

CMOS cameras offer several advantages: low cost, low power, easy image region of interest selection, on-chip characteristics and high frame rate. This makes them a natural fit for security cameras, videoconferencing, wireless handheld devices, toys and biometrics. Standard and cheap CMOS cameras frequently use rolling shutter sensors. In this acquisition mode, the pixels are not all exposed simultaneously but row by row starting from the top and with a time delay defined by the sensor technology. Rolling shutter mode enables adequate exposure time without reducing the frame rate by overlapping exposure and readout steps. As result, the fill factor (percentage of the pixel array sensitive to light) and the signal to noise ratio are improved by reducing the number of in-pixel transistors. The drawback of a rolling shutter camera is that it distorts images of moving objects as seen on Figure 1. This distortion may represent a major obstacle in tasks such as localization, reconstruction or default detection.

Previous work has addressed perspective projection modeling of rolling shutter cameras [1, 2, 3, 6, 7]. In [7]

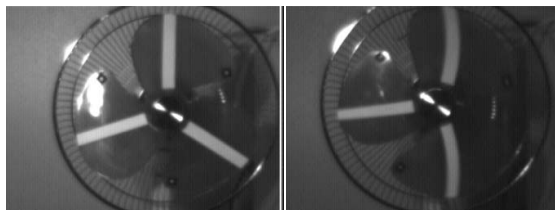


Figure 1. Rolling shutter effects on moving object images (straight lines seems curved on the right image where the object is rotating).

a projection model is presented assuming small motion during the acquisition of one image. In the case of fronto-parallel motion equations are similar to those of Crossed-Slits cameras [10]. A more exact model is described in [2]. This model is then used to simultaneously estimate the pose and the velocity of a moving object with known geometry from a single rolling shutter image. Another approach, presented in [6], addressed the compensation of rolling shutter effects assuming a planar motion model. Spatio-temporal triangulation with classical cameras generally involves matching image sequences from multiple cameras as in [5] where complex image processing at high frame rate is needed for surface representation.

In this paper, we show how one pair of rolling shutter images enables the computation of both structure and velocity of rigid moving objects. The heavy constraint involving a-priori knowledge about the scene geometry assumed in previous work is removed. The method uses only image point correspondences. The main difficulty is that matched points are not necessarily viewed at the same time. Thus, classical triangulation methods can not be used. We present a spatio-temporal triangulation method which takes into account both image scanning speed and object motion (through its 6 D.O.F. velocity parameters).

The work presented here aims to make the use of rolling shutter cameras of a more broad interest. Indeed, structure from motion algorithms usually requires images from sufficiently different angles of view in order to avoid poor tri-

angulation. If an object is passing very rapidly through a tight field of view, one may imagine that the system will not always be able to grab several images of this object in order to recover its kinematics. Our method offers the possibility to recover the motion with one snapshot. From another point of view, actual computer vision applications request more and more fast methods to satisfy real time requirement. Even by increasing algorithm efficiency and computer speed, we still face the problem of bottleneck in the transmission line. Our method tends to reduce the amount of data needed to recover the motion because one well exploited rolling shutter image contains the same information about the motion than a long sequence.

In section 2, we briefly recall the basic equation of a rolling shutter projection model. In section 3, the stereo projection equations are presented. In section 4, we describe the structure and velocity computation method. Experiment results with both synthetic and real data are presented in section 5.

2. Rolling Shutter Camera Projection Model

In a CMOS camera operating in rolling shutter mode, the sensor pixels are exposed sequentially starting at the top and proceeding row by row to the bottom. The readout process proceeds in exactly the same fashion and the same speed with a time delay after the reset (this defines exposure time). The benefit of rolling shutter mode is that exposure and readout are overlapping. This enables full frame exposures without reducing the frame rate. Each row in the image has the same integration time, however the starting and ending time of integration are shifted in time as the image is scanned (rolled). If an observed object is moving during the integration time, some artefacts may appear and its image is distorted.

Assume that an object modeled by a set of n points $\mathbf{P}_i = [X_i, Y_i, Z_i, 1]^T$, undergoing a motion with instantaneous angular velocity Ω around an instantaneous axis of unit vector $\mathbf{a} = [a_x, a_y, a_z]^T$, and instantaneous linear velocity $\mathbf{V} = [V_x, V_y, V_z, 1]^T$, is observed by a rolling shutter camera at time t_0 . Because the scanning is very fast, one can reasonably assume constant values for \mathbf{V} , \mathbf{a} and Ω during one image acquisition. Denoting \mathbf{R} and \mathbf{T} the instantaneous object pose at t_0 and τ the time needed for the integration of one image line (image scanning speed in lines per second), the point \mathbf{P}_i is projected onto $\mathbf{m}_i = [u_i, v_i, 1]^T$ after a delay $t_i = \tau v_i$. As seen in [2], during this delay, the object performs a rotation $\delta\mathbf{R}_i$ and a translation $\delta\mathbf{T}_i$. The projection of \mathbf{P}_i can thus be expressed up to an arbitrary scale factor s as follows:

$$s\mathbf{m}_i = \mathbf{K} \begin{bmatrix} \mathbf{R}\delta\mathbf{R}_i & \mathbf{T} + \delta\mathbf{T}_i \end{bmatrix} \mathbf{P}_i \quad (1)$$

with:

$$\delta\mathbf{R}_i = \mathbf{a}\mathbf{a}^T (1 - \cos(\tau v_i \Omega)) + \mathbf{I} \cos(\tau v_i \Omega) + [\mathbf{a}]_{\times} \sin(\tau v_i \Omega) \quad (2)$$

and:

$$\delta\mathbf{T}_i = \tau v_i \mathbf{V} \quad (3)$$

where \mathbf{I} is the 3×3 identity matrix, $[\mathbf{a}]_{\times}$ is the skew-symmetric matrix associated to \mathbf{a} and \mathbf{K} contains the classical intrinsic parameters of a pinhole camera model. Note that \mathbf{V} is expressed in the camera frame.

Equation (1) is the expression of the projection of a 3D point from a moving solid object using a rolling shutter camera with respect to object pose, object velocity and the parameter τ . Note that it contains the unknown v_i on its two sides. This is due to the fact that coordinates of the projected point on the image depend on both the kinematics of the object and the imager sensor scanning speed.

3. Stereo with Rolling Shutter Cameras

We consider a moving object observed by a pair of rolling shutter cameras. The left camera is chosen as the reference frame. The points \mathbf{P}_i are then expressed in this frame. We denote \mathbf{R} and \mathbf{T} the right-to-left camera frame transformation. As the object moves and the cameras' shutter scans through the images, each scanned line does not see the object at the same position as shown in Figure 2. This is expressed by the following projection equations of a rolling shutter pair:

$$\mathbf{m}_i \sim \mathbf{K} \begin{bmatrix} \delta\mathbf{R}_i & \delta\mathbf{T}_i \end{bmatrix} \mathbf{P}_i \quad (4)$$

$$\mathbf{m}'_i \sim \mathbf{K}' \begin{bmatrix} \mathbf{R}\delta\mathbf{R}'_i & \mathbf{T} + \delta\mathbf{T}'_i \end{bmatrix} \mathbf{P}_i, \quad (5)$$

which include as unknowns the 3D point coordinates \mathbf{P}_i , the pose and velocity of the object, represented by the rotation axis \mathbf{a} and speed Ω , and the translational speed \mathbf{V} . These unknowns are linked to the object pose through the Rodrigues formula:

$$\begin{aligned} \delta\mathbf{R}_i &= \mathbf{a}\mathbf{a}^T (1 - \cos(\tau v_i \Omega)) + \mathbf{I} \cos(\tau v_i \Omega) + [\mathbf{a}]_{\times} \sin(\tau v_i \Omega) \\ \delta\mathbf{T}_i &= \tau v_i \mathbf{V} \\ \delta\mathbf{R}'_i &= \mathbf{b}\mathbf{b}^T (1 - \cos(\tau' v'_i \Omega)) + \mathbf{I} \cos(\tau' v'_i \Omega) + [\mathbf{b}]_{\times} \sin(\tau' v'_i \Omega) \\ \delta\mathbf{T}'_i &= \tau' v'_i \mathbf{V}' \end{aligned} \quad (6)$$

Velocity parameters $[\mathbf{b}, \mathbf{V}']$ are obtained by the kinematic twist transformation matrix of $[\mathbf{a}, \mathbf{V}]$ from the left camera frame to the right camera frame given as:

$$\begin{bmatrix} \mathbf{V}' \\ \mathbf{b} \end{bmatrix} = \begin{bmatrix} \mathbf{R} & [\mathbf{T}]_{\times} \mathbf{R} \\ \mathbf{0}_3 & \mathbf{R} \end{bmatrix} \begin{bmatrix} \mathbf{V} \\ \Omega \mathbf{a} \end{bmatrix} \quad (7)$$

where $\mathbf{0}_3$ is the 3×3 null matrix. This forms the basis for the stereo triangulation algorithm described in the next section.

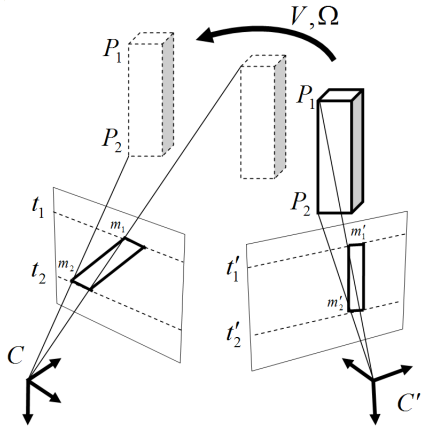


Figure 2. Stereo projection in rolling shutter mode.

4. Structure and Velocity Computation from a Rolling Shutter Pair

4.1. Problem Statement

Standard triangulation proceeds by computing each point independently. Noting \mathbf{M}_i and \mathbf{M}'_i the projection matrices, this is done by solving for \mathbf{P}_i the following linear system with each image point pair using singular value decomposition:

$$\begin{bmatrix} [\mathbf{m}_i]_{\times} \mathbf{M}_i \\ [\mathbf{m}'_i]_{\times} \mathbf{M}'_i \end{bmatrix} \mathbf{P}_i = \mathbf{0} \quad (8)$$

For the problem at hand, triangulation from each pair of image points $(\mathbf{m}_i, \mathbf{m}'_i)$ would give a misplaced point \mathbf{P}_i because \mathbf{m}_i and \mathbf{m}'_i are images, at different times, of a moving 3D point. In fact, we can equivalently say that \mathbf{m}_i and \mathbf{m}'_i are images of different 3D points. The idea here is to take into account the movement of the 3D points in the triangulation equations. This is achieved by introducing the velocity parameters which are common to all the points \mathbf{P}_i (assuming constant speed during the acquisition of one image as stated previously) through the projection matrices \mathbf{M}_i and \mathbf{M}'_i which become:

$$\begin{aligned} \mathbf{M}_i &= \mathbf{K} \begin{bmatrix} \delta \mathbf{R}_i & \delta \mathbf{T}_i \end{bmatrix} \\ \mathbf{M}'_i &= \mathbf{K}' \begin{bmatrix} \mathbf{R} \delta \mathbf{R}'_i & \mathbf{T} + \delta \mathbf{T}'_i \end{bmatrix} \end{aligned} \quad (9)$$

Equation system (8) can now be seen as a cost function which is nonlinear in parameters $[\mathbf{P}_i, \mathbf{a}, \Omega, \mathbf{V}]$ and needs to be over-determined. Triangulation must thus be performed for all the points at once by solving a non linear optimization problem consisting in minimizing the algebraic distance defined by equation (8).

An alternative method consists in using the reprojection error as a criterion. We define a cost function \mathcal{C} which pe-

nalizes the distance between the actual image points $\hat{\mathbf{m}}_i$ and the predicted ones $\mathbf{m}_i(\psi)$ where ψ is a vector that contains all the parameters $\mathbf{P}_1, \dots, \mathbf{P}_n, \mathbf{a}, \Omega$ and \mathbf{V} :

$$\mathcal{C}(\psi) = \sum_{i=1}^n \left(\|\hat{\mathbf{m}}_i - \mathbf{m}_i(\psi)\|^2 + \|\hat{\mathbf{m}}'_i - \mathbf{m}'_i(\psi)\|^2 \right) \quad (10)$$

The latter approach gives better results. We solve this nonlinear least squares problem as described in the next section.

4.2. Nonlinear Optimization

The solution proceeds in two steps. First, we compute a suboptimal initial estimate of the 3D points neglecting rolling shutter effects. Second, we optimize the cost function defined above starting from the suboptimal initial shape and a null velocity vector.

For the first step, we make the assumption that the classical hypothesis of neglecting the effect of rolling shutter holds to some extent. This is a classical triangulation problem, for which a closed-form solution is provided in [4]. Given the above computer 3D points, the non-linear least square optimization can be started.

The cost function in (10) involves $7 + 3n$ unknowns. Directly using a nonlinear least squares routine to refine the cost would result in a computationally expensive procedure, whose complexity cubic in the number n of points. This can be achieved with a complexity linear in the number n of points basing on the usual way one uses in bundle block adjustment (See [8] for example).

4.3. Degenerate Configuration

The success of structure and velocity computation is based on the fact that epipolar geometry is not satisfied with the initial reconstructed points unless the motion is taken into account. Indeed, when the object undergoes a general motion, the points can not remain in their respective epipolar planes during this movement. Can we deform the object such that all the points \mathbf{P}_i satisfy the epipolar geometry with a speed parameter different from the real one? If yes a degenerate configuration occurs. In which case do all the points \mathbf{P}_i remain on their respective epipolar planes during the motion? Without a formal demonstration, we show on Figure 3 that this is possible only if the object undergoes a pure translation with a velocity vector exactly parallel to the baseline linking the two camera centres \mathbf{CC}' . Indeed, the epipolar planes π_i are not parallel and all intersect on the line \mathbf{CC}' . During a pure translation, all the points \mathbf{P}_i move along parallel lines. Thus, the only motion which enables to satisfy epipolar geometry is a translation along \mathbf{CC}' . In some situations this degeneracy can be removed if some knowledge about the shape is available (planar parts of the object, known angles, etc).

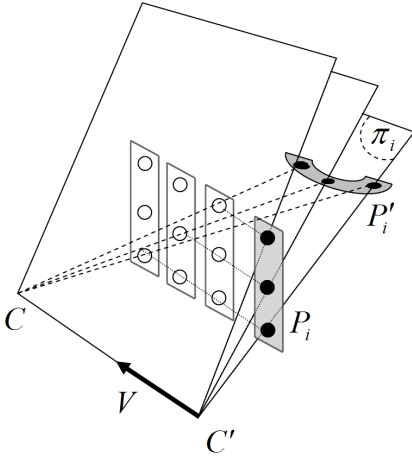


Figure 3. Degenerate configuration: for a pure translation parallel to the baseline, the spatio-temporal triangulation may give an imaginary shape P'_i with a null velocity instead of the real shape P_i with the velocity \mathbf{V} .

5. Experimental Results

The presented method was tested with both synthetic and real data. Experiment setup and obtained results are presented below.

5.1. Experiments with synthetic data

To test the reconstruction and the velocity measurement accuracy, a rolling shutter stereo rig simulator was developed. Intrinsic, extrinsic and shutter parameters are then known. A synthetic object of a known geometry is moved with known velocity parameters yielding ground truth values for the experiments. The 3D points are projected on the two images using the rolling shutter model for the left one and a classical model for the right one. Obtained image point correspondences are then used to reconstruct the 3D points and to compute the velocity.

Simulated cameras resolution was 1024×1024 pixels and the focal lengths varied from 6 mm to 12 mm. The scan parameter τ varied from 50 to $100 \mu\text{sec}$. A virtual object representing a 3D grid was used with dimensions varying from 20×20 cm to 40×40 cm was used. A set of 100 image pairs corresponding to different velocity parameters was generated. The depth of the points varied from 0.6 m to 1.20 m. The orientation of vectors \mathbf{V} and \mathbf{a} was varying in all the space directions. The magnitude of \mathbf{V} was in the range $[0, 10]$ m/sec. The magnitude of $\mathbf{\Omega}$ was in the range $[0, 20]$ rad/sec.

A noise was added to both calibration parameters and image points. The 6 noise levels used in the tests are defined in Table 1. For each noise level and for each image pair, 100

Noise level	0	1	2	3	4	5
α_u, α_v (%)	0.0	0.2	0.4	0.6	0.8	1.0
u_0, v_0 (pixels)	0.0	1.0	2.0	3.0	4.0	5.0
u_i, v_i (pixels)	0.0	0.2	0.4	0.6	0.8	1.0

Table 1. Noise levels used for synthetic data experiments: values for intrinsic parameters α_u, α_v , u_0, v_0 are the values added or subtracted from real parameters. Values for image point measurements u_i, v_i are standard deviations of the added gaussian noise.

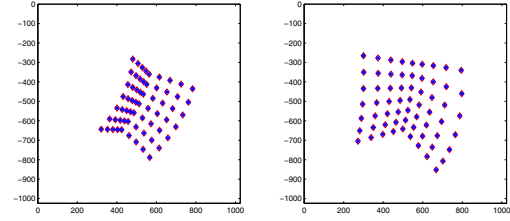


Figure 4. An example of left and right synthetic image points.

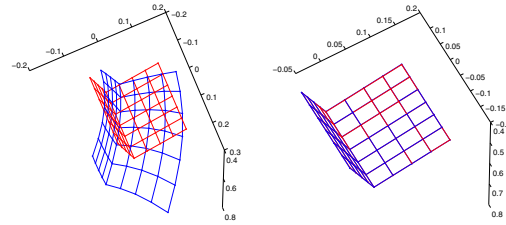


Figure 5. Reconstruction results obtained with the previous synthetic image pair: reference points (in red), initial solution (in blue on the left graph) and final solution (in blue on the right graph).

random noise vectors were added to the exact data. The structure and velocity computation algorithm was then run. Figure 4 shows an example of a synthetic image pair and Figure 5 shows the corresponding reconstructed shape.

Obtained results are summarized in Figure 6, Figure 7 and Figure 8. Note that only 2 cases where the algorithm converged toward a false local minimum occurred.

5.2. Experiments with real data

A 3D known object was mounted on a rotating mechanism which provides ground truth values for velocity. The stereo rig was calibrated using the method described in [9]. Several image pairs were taken from different angles of view. The structure and the velocity of the object were then computed using the algorithm presented above.

Two examples of image pairs and corresponding 3D re-

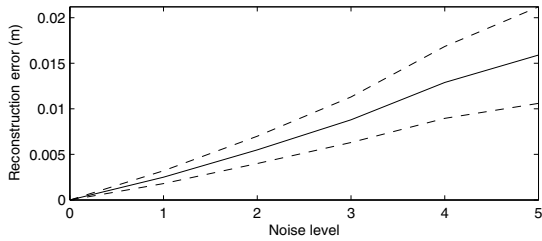


Figure 6. Reconstruction error with synthetic data experiments.

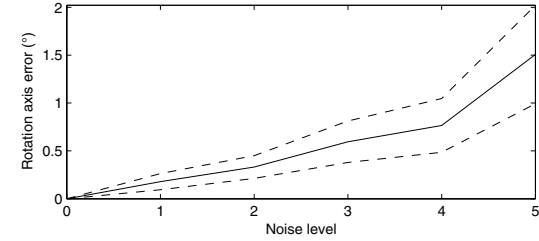
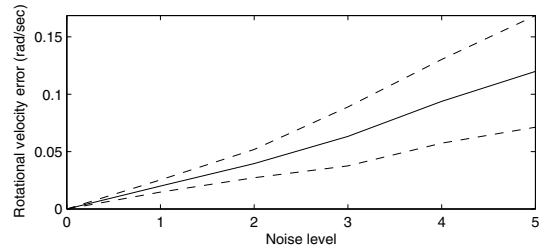


Figure 8. Rotational velocity error and rotational axis error with synthetic data experiments.

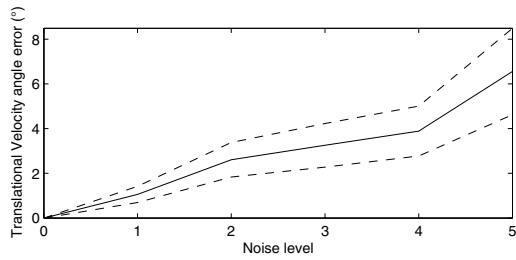
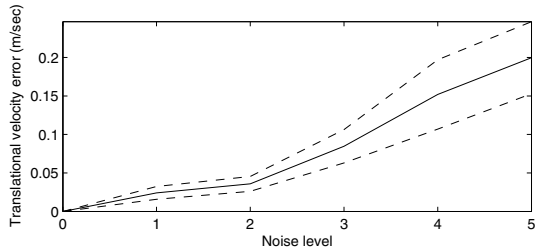


Figure 7. Translational velocity vector error with synthetic data experiments (magnitude and direction).

constructions are given on Figure 9 and Figure 10 respectively. Reconstruction error and velocity computation results obtained for 3 sequences taken from different view angles are presented on Figure 11. Results show that performances are in the range obtained with synthetic data. Note that for very slow motion, the rotation axis is very badly estimated because the magnitude of the rotation vector is small. Thus, axis direction estimating is very sensitive to noise. Fortunately, this does not affect the precision of reconstruction too much because image scanning is very fast in comparison to such a slow motion and thus rolling shutter effects are not very important. In the same way, when the motion is very fast, the precision of translational speed diminishes and reconstruction error augments due to rolling shutter effects. These problems can easily be fixed by choosing an adequate value for parameter τ according to the range of speeds one aims to measure.

Figure 12 shows how rolling shutter image points can be rectified by reprojecting the reconstructed points using a classical projection model.

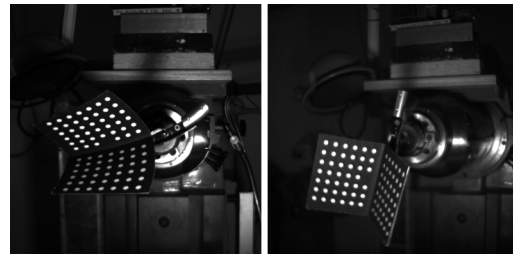
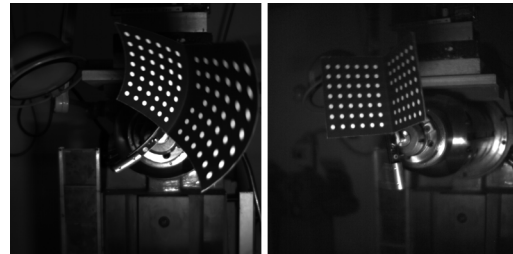


Figure 9. An example of 2 image pairs used in real data experiments.

6. Conclusion and Perspectives

A method for both structure and kinematics computation from a single rolling shutter image pair was presented. It is based only on point correspondences and takes advantage of motion artefacts in the image. The method enables rolling shutter effects compensating and reduces the amount of data needed in structure from motion algorithms. Doing so, it makes rolling shutter cameras of a more broad interest.

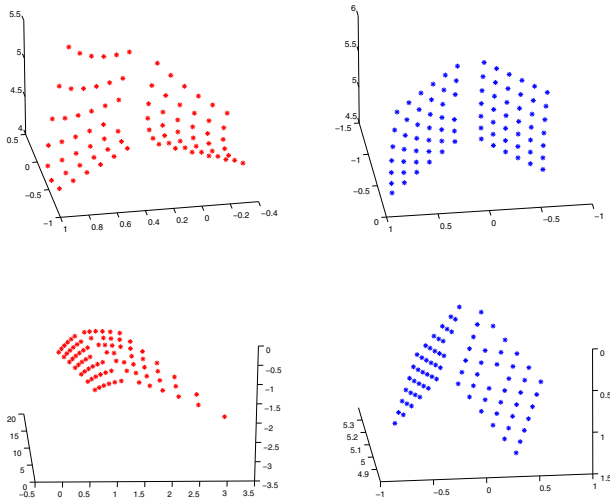


Figure 10. Reconstruction from image pair 1 (top graphs) and from image pair 2 (bottom graphs): left graphs show initial structures and right graphs show refined structure.

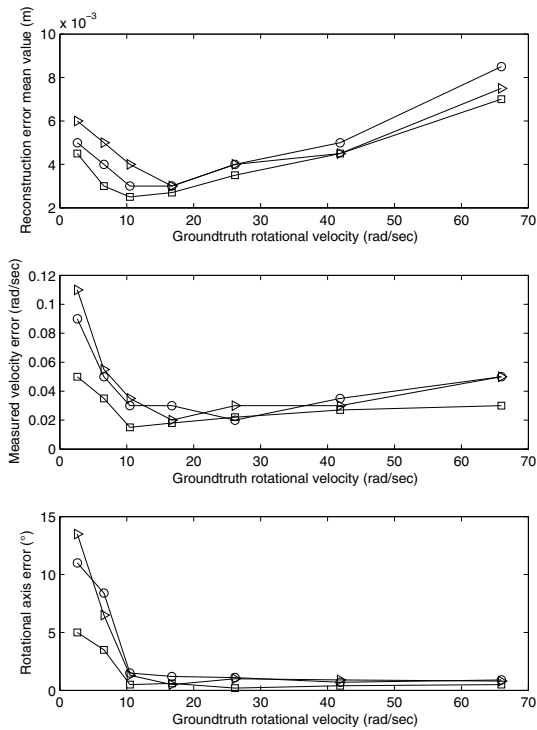


Figure 11. Experiment results with real data sequences taken from three different view points.

Our current work focuses on the matching problem. A RANSAC approach can be used. However, for very fast

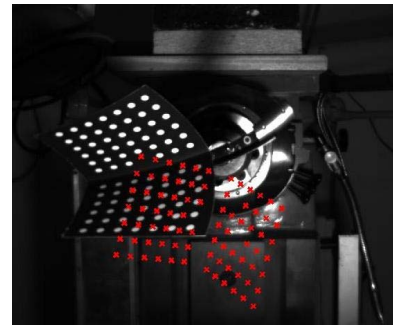


Figure 12. Rolling shutter effect compensation: red points represent what a classical camera would see.

motion, local effects of rolling shutter can affect correlation results during the candidate match generation step. For instance, effects on Harris points should be investigated.

References

- [1] O. Ait-Aider, N. Andreff, J. M. Lavest, and P. Martinet. Exploiting rolling shutter distortions for simultaneous object pose and velocity computation using a single view. In *Proc. IEEE International Conference on Computer Vision Systems*, New York, USA, January 2006. 1
- [2] O. Ait-Aider, N. Andreff, J.-M. Lavest, and P. Martinet. Simultaneous object pose and velocity computation using a single view from a rolling shutter camera. In *ECCV (2)*, pages 56–68, 2006. 1, 2
- [3] O. Ait-Aider, A. Bartoli, and N. Andreff. Kinematics from lines in a single rolling shutter image. In *CVPR, 2007*. 1
- [4] R. I. Hartley and P. Sturm. Triangulation. *Comput. Vis. Image Underst.*, 68(2):146–157, November 1997. 3
- [5] N. Jan and A. Yiannis. Spatio-temporal stereo using multi-resolution subdivision surfaces. *Int. J. Comput. Vision*, 47(1-3):181–193, 2002. 1
- [6] C.-K. Liang, L.-W. Chang, and H. H. Chen. Analysis and compensation of rolling shutter effect. *IEEE Transactions on Image Processing*, 17(8):1323–1330, 2008. 1
- [7] M. Meingast, C. Geyer, and S. Sastry. Geometric models of rolling-shutter cameras. In *Proc. of the 6th Workshop on Omnidirectional Vision, Camera Networks and Non-Classical Cameras*, Beijing, China, October 2005. 1
- [8] B. Triggs, P. F. McLauchlan, R. I. Hartley, and A. W. Fitzgibbon. Bundle adjustment—a modern synthesis. In *Proc. of the International Workshop on Vision Algorithms: Theory and Practice*, pp. 298–372, London, UK, 1999. 3
- [9] Z. Zhang. A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(11):1330–1334, 2000. 4
- [10] A. Zomet, D. Feldman, S. Peleg, and D. Weinshall. Mosaicing new views: The crossed-slits projection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(6):741–754, 2003. 1